Automatic Extraction of Data from Bar Charts

Rabah A. Al-Zaidy
The Pennsylvania State University
University Park, PA
alzaidy@psu.edu

C. Lee Giles
The Pennsylvania State University
University Park, PA
giles@ist.psu.edu

ABSTRACT

Scientific charts are an effective tool to visualize numerical data trends. They appear in a wide range of contexts, from experimental results in scientific papers to statistical analyses in business reports. The abundance of scientific charts in the web has made it inevitable for search engines to include them as indexed content. However, the queries based on only the textual data used to tag the images can limit query results. Many studies exist to address the extraction of data from scientific diagrams in order to improve search results. In our approach to achieving this goal, we attempt to enhance the semantic labeling of the charts by using the original data values that these charts were designed to represent. In this paper, we describe a method to extract data values from a specific class of charts, bar charts. The extraction process is fully automated using image processing and text recognition techniques combined with various heuristics derived from the graphical properties of bar charts. The extracted information can be used to enrich the indexing content for bar charts and improve search results. We evaluate the effectiveness of our method on bar charts drawn from the web as well as charts embedded in digital documents.

Keywords

Information extraction, scientific chart understanding, web search

1. INTRODUCTION

Scientific charts have wide presence not only as images in the web, but also as embedded figures in PDF documents. Main search engines nowadays include figures in search results. However, indexed content for charts, and documents containing them, rely mainly on the metadata textual tags. By not including the actual information these charts represent in the query process, search engines may overlook many valuable query results. Thus, enriching the indexing content for both documents and images based on the chart's content, provides an additional dimension to search improvement. Many studies aim to extract various types of information from scientific charts including bar charts. Automatically extracting the data values from bar charts can effectively enhance their semantic labeling. This additional information for charts can allow various analyses for search tools. Additionally, it can further assist the application of information retrieval and knowledge discovery techniques. In this paper, we describe a system that targets bar charts specifically. The method automatically extracts text and graphical components from the charts and combines the results to infer the original data values of the chart.

K-CAP 2015 October 07 - 10, 2015, Palisades, NY, USA

ACM 978-1-4503-3849-3/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2815833.2816956.

2. RELATED WORK

The problem of understanding scientific charts has been addressed in various studies. In [1] they describe a method to automatically annotate each text in the chart with a semantic role, e.g., axis labels, caption, etc. The charts along with their annotated semantics are used to build an index for their web-based diagram search engine described in [2]. Although the annotator extracts textual and graphical components for the role labeling, the method does not provide further processing to include the original data values of the chart as well. In other approaches the data values are extracted and recovered but only for black and white or grayscale charts, as in [3] and [7]. A study relevant to our work is [9], where they propose a system to analyze and redesign charts. In their approach they use a method to infer the original chart data values upon extracting graphical and textual components. The graphical components are extracted automatically, however the user is required to specify regions of the chart where textual components are located. Another similar study that automates the extraction and data recovery process is [6]. In their approach they extract data values of the chart based on a mapping between text and graphic components. Each chart is then represented as an XML file containing data values and text information of the chart. The method is applied to images found in the web. It applies a model-based approach to detect the chart type and extract the components. However, the bar chart data extraction method which they describe in further detail in [5], handles only single-series bar charts, where all bars are assumed to be the same color.

3. METHOD

Our method follows the pipeline shown in Figure 1. The system comprises of three main modules: graphical-component extraction module, text extraction module, and the data inference module. The system takes a 2-dimensional bar

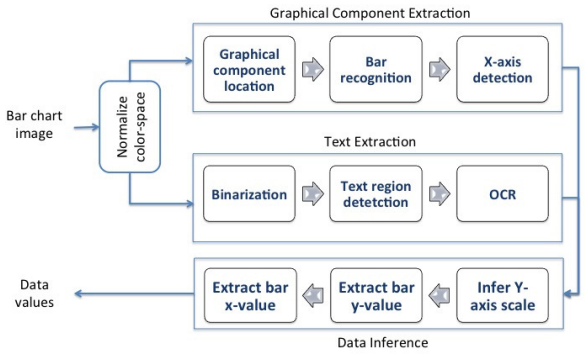

Figure 1: Data Extraction System Overview

chart in image format as input and regenerates the chart as output. The following assumptions are made regarding the charts:

- Charts are 2-dimensional charts.
- The fill color for the bars is a solid color, rather than a pattern.
- The y-axis follows a linear scale, rather than logarithmic.
- Y-axis alignment is to the left, x-axis alignment is to the bottom of the chart.

Following these assumptions the system modules extract bar data automatically. The system initially applies some pre-processing steps to the chart image, such as color normalization and noise filtering. Algorithms for text recognition in graphics typically apply noise removal filters to the image first [4]. For scientific charts, however, it is uncommon to have highly noisy images [10] as compared to non-chart images. Additionally, noise removal filters, e.g. bilateral filter, while computationally costly, provided no significant improvements when applied to charts that were extracted from PDF documents. Thus, we do not apply any noise filtering step. The only pre-processing step in our method is changing the color-space prior to passing the chart to the extraction modules.The text extraction module requires a grayscale version of the chart. For the graphical component extraction the image is converted to the Lab color space. This is due to the fact that our method is expected to apply to colored charts just as effectively as black and white ones. The images in PDF documents are recovered in raster format, where each pixel is specified by its RGB value. Processing a chart in the RGB color scale proposes the following challenges:

- A single bar can have very light color that is close in RGB value to that of the background. This makes it less distinguishable as a distinct graphical component in the chart.
- In multiple-series bar charts where bars can share an adjacent edge, adjacent bars can be mistaken to be the same graphical component if their RGB colors do not highly contrast.

These challenges are far less present when the recognition of components is applied to in image in the LAB color space. The LAB color space is more perceptually uniform than the RGB space. This implies that the amount of change in the color values corresponds to a similar amount of change in human perception. This makes it effective in extracting information that is distinguishable based on visual effects, such as chart components. In order to extract meaningful information from the chart, two basic tasks must be performed. First, extraction of the graphical components, i.e. the bars and axes. The other task is the retrieval of texts that label these graphical components. Additional processing of the extracted information is required to obtain the data values of the chart. The methods proposed to achieve all these tasks are described in the remainder of this section.

3.1 Graphical Component Extraction

Once the bar is converted to the Lab space, the graphical component extraction module is responsible for identifying the main chart component, the bars. For accurate data extraction, the bar extraction method must first: correctly identify each bar distinctly, and secondly have high recall, i.e. identify as much bars as possible (preferably all of them). In order to recover the bars, our algorithm follows a similar approach to the one presented in [9]. We perform connected component labeling to the image, however our method uses the Lab color space to distinguish components. For the connected component method, neighboring pixels who differ in color are labeled as different components. To compare colors in the Lab space we use the delta-E 95 distance equation. Upon experimentation the threshold of 7 was found suitable. Once the connected components are recovered, we identify the bar components by using heuristics derived from the graphical properties of the bars. The following properties are used to distinguish a bar component:

- A bar fills its bounding box with ratio greater than 90%.
- The color of any pixels inside the bar is different than the color of all pixels within 2 to 3 pixels distance outside the bar edges.

Each bar is defined by its location and height in pixels. In order to identify the x-axis we use the conjecture that the x-axis is the horizontal line that all bars have an edge on. We use a histogram of the location of the base of the bars to recover a common horizontal coordinate among the horizontal edges of all the bars. This is defined as the x-axis.

3.2 Text Component Extraction

The texts associated with the graphical components of the chart are key elements to recover the original values represented by the chart components. Labels for x and y-axes, and each of the bars, in addition to the numerical values marking the scale of the y-axis. In this section we describe our proposed method to automatically extract these pieces of valuable information. The method involves two basic steps as following the method in [4]. The first step is to automatically locate text regions in the image. The next step is to apply OCR recognition to these text components.

3.2.1 Identification of Text Regions in a Chart

To identify the text regions in a chart we follow similar steps to the method proposed in [4]. The image is already

binarized in step A above. We remove components whose area size is greater than that of a typical character size. Next, in order to identify which letters represent a single word, the letters are subjected to an isotropic dilation with a small window size. This will close the small gaps between pixels that are close to one another just enough to be adjacent. Then we apply connected component labeling to the dilated image. This labeling will label entire words as one component. To maintain image quality for the OCR, once the locations of the text regions are specified, the text region blocks passed to the OCR step are from the image before dilation rather than the dilated ones. If a text region's width is smaller than its height, it is vertical and most likely to be the name of the y-axis.

3.2.2 *Extracting Text and Numerical Values*

The tesseract OCR is used to recognize the nominal or numerical values of each text region. The results are filtered out for any text regions that produce empty spaces or only punctuation marks. For instance, some stray pixels can be identifies as texts and return the '.' character. Finally, the recovered texts are represented as their nominal or numerical values along with their locations.

Algorithm 1: Data inference algorithm for extraction of chart numerical values and axes names.

```
Input  : x-axis, y-axis, B, T
Output: Data values (data:name; data:value),
        Y_name, X_name
if T_i:y below x-axis then
  | X  T_i
end
Assign elements of X to X_name and X_labels
based on horizontal alignment
if T_i:x to left of y-axis then
  | Y  T_i
end
Assign elements of Y to Y_name and Y_labels
based on vertical alignment
for T_i  Y_labels do
  | Scale_i = T_{i+1}:y - T_i:y / T_{i+1}:value - T_i:value
end
□ = median(Scale);
Set data_i:value  B_i:height * □;
Set data_i:name  X_labels_i where:;
X_labels_i:x is closest to B_i:x
return : set of data pairs (data:name; data:value)
```

3.3 Chart Data Inference

In this step we extract the chart data by applying an inference process. The name of the bar is the text identifying the bar, which is typically located under the bar. The value of the bar is the y-axis value that corresponds to the highest point of the bar. To extract this data pair we follow steps similar to those in [9]. Three main values must be recovered to accurately obtain the bar values. These are: the y-axis scale values, the data-per-pixel ratio, and the bar name, i.e. x values. Algorithm 1 shows the steps of the inference method. As input, it requires the location of the x-axis and y-axis, which have been previously located, the set of bars B, and the extracted text strings T. The y-axis is determined to be the area left to the left-most bar. The x-axis location has been extracted in the previous graphical component extraction step. Each graphical component in B_i contains 3 values. The x and y coordinates and the height of the bar, denoted B_i:x, B_i:y, and B_i:height, respectively. The text strings in T also contain 3 values. The x and y coordinates of the text region and the value of the string, denoted T_i:x, T_i:y, and T_i:value, respectively. The first value to infer, the y-scale, relies on both the correct location of the texts describing the scale and the result of the OCR. To specify the texts that correspond to this values, we apply two heuristics: the y-axis labels are to left of the left-most bar, and the are vertically aligned. Those texts that satisfy these assumptions are sorted and used for the next step. The next step is identifying the pixel-per-data ratio. For this, we calculate the difference between the y-labels in pixel and divide it by the difference in the numerical values of the y-label texts. The bar values are thus, their heights in pixels multiplied by the pixel-per-data ratio. The Nominal value for each bar is the value of the text located below the bar. To identify the values we extract the text regions who are located below the x-axis and are horizontally aligned.

	Percentage of bars extracted
PDF embedded bar charts	87%
ATLAS charts	95%
Web images	77%

Table 1: Extraction accuracy

4. EXPERIMENTS

In order to evaluate the effectiveness of our method we tested its accuracy on bar charts found in the web and charts embedded in PDF documents. We examine the results of the system on a total of 18 bar charts, 8 are extracted from PDF documents, 5 are from the data set of web images provided by [9] in their ReVision tool. We also experimented with 5 bar charts that were generated from scientific chart generation tools such as Chartbuilder by [8].

Table 1 shows the percentage of correctly recovered bars in a bar chart. The method recovers, on average, 87% of bars in the first class of images, PDF embedded charts. However, due to inaccuracies of the OCR tool the y-scaling of almost 20% of these charts was not calculated correctly, even though the bars were identified and located. The highest accuracies is for the bar charts obtained from the Atlas database. That is due to the image quality and the relative simplicity of the bars. Mostly the missed bars were very small in size. Similar to the previous class the y-scale evaluated correctly for most charts except when OCR results are incorrect. The bar extraction obtained good results for web images, however due to the quality of the images, the OCR did not detect the y-axis label values for over 80% of the charts. For this type of images, more enhancement to the text region is required to obtain final values for the chart data. Figure 2 shows a sample input chart compared to the resulting chart thats is reconstructed based on extracted values.

Some limitations to this approach are noted. Since we do not parse legends, that accounts for missing information for the nominal value of the bar. However, since the colors of the bars are distinguished with high accuracies, the parsing of legends can be a feasible extension to this work. Addition-

ally, further text enhancement techniques can be applied to increase the accuracy of the results of the OCR tool. This can improve the accuracy of the y-scale extraction and the axes name extraction.

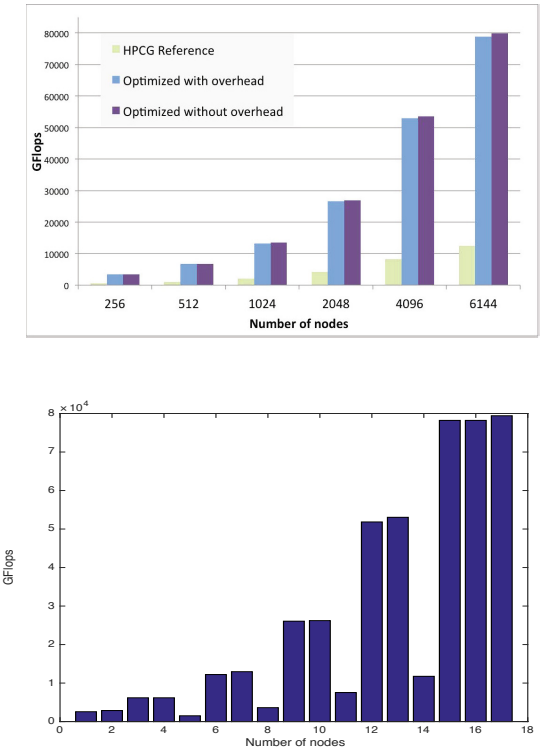

Figure 2: A sample bar chart extracted from a random PDF file and the same chart reconstructed using the extracted data values.

5. CONCLUSION

Although many studies address the problem of understanding scientific charts, very few specifically address the bar chart class. In this paper we apply image processing techniques to extract data from bar charts embedded in digital documents to enhance that is useful for semantic labeling of documents and images. We describe a system to extract graphical and text components from bar charts to reproduce the original data values of the chart. Experimental evaluation demonstrates the effectiveness of our method, with bar recovery accuracies of up to 87% for PDF embedded bar charts. The extracted values can be further used in future studies for development of domain-specific knowledge discovery applications or enhancing query and snippet generation.

6. ACKNOWLEDGEMENTS

The authors would like to thank Jian Wu, for his many insightful comments.

7. REFERENCES

[1] S. Z. Chen, M. J. Cafarella, and E. Adar. Searching for statistical diagrams. Frontiers of Engineering, National Academy of Engineering, pages 69–78, 2011.

[2] Z. Chen, M. Cafarella, and E. Adar. Diagramflyer: A search engine for data-driven diagrams. In Proceedings of the 24th International Conference on World Wide Web Companion, pages 183–186. International World Wide Web Conferences Steering Committee, 2015.

[3] D. Chester and S. Elzer. Getting computers to see information graphics so users do not have to. In Foundations of Intelligent Systems, pages 660–668. Springer, 2005.

[4] L. A. Fletcher and R. Kasturi. A robust algorithm for text string separation from mixed text/graphics images. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 10(6):910–918, 1988.

[5] W. Huang, C. L. Tan, and W. K. Leow. Model-based chart image recognition. In Graphics Recognition. Recent Advances and Perspectives, pages 87–99. Springer, 2004.

[6] W. Huang, C. L. Tan, and W. K. Leow. Associating text and graphics for scientific chart understanding. In Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on, pages 580–584. IEEE, 2005.

[7] Y. Liu, P. Mitra, C. L. Giles, and K. Bai. Automatic extraction of table metadata from digital documents. In Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries, pages 339–340. ACM, 2006.

[8] Quartz. Atlas, by quartz. `http://atlas.qz.com/`, 2015.

[9] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer. Revision: Automated classification, analysis and redesign of chart images. In Proceedings of the 24th annual ACM symposium on User interface software and technology, pages 393–402. ACM, 2011.

[10] N. Vassilieva and Y. Fomina. Text detection in chart images. Pattern Recognition and Image Analysis, 23(1):139–144, 2013.